Information Processing and Management 56 (2019) 565–583

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/infoproman

Online visibility of software-related web sites: The case of biomedical text mining tools

Gael Pérez-Rodríguez[a,b,c], Martín Pérez-Pérez[a,b,c], Florentino Fdez-Riverola[a,b,c], Anália Lourenço[a,b,c,d,*]

[a] Department of Computer Science, University of Vigo, ESEI, Campus As Lagoas, 32004 Ourense, Spain
[b] The Biomedical Research Centre (CINBIO), Campus Universitario Lagoas-Marcosende, 36310 Vigo, Spain
[c] SING Research Group, Galicia Sur Health Research Institute (IIS Galicia Sur), SERGAS-UVIGO, Vigo, Spain
[d] Centre of Biological Engineering (CEB), University of Minho, Campus de Gualtar, 4710-057 Braga, Portugal

ARTICLE INFO

Keywords:
Online visibility
Literature
Data mining
Explanatory and predictive models
Software tools

ABSTRACT

Internet, in general, and the WWW, in particular, have become an immediate, practical means of introducing software tools and resources, and most importantly, a key vehicle to attract the attention of the potential users. In this scenario, content organization as well as different development practices may affect the online visibility of the target resource. Therefore, the careful selection, organization and presentation of contents are critical to guarantee that the main features of the target tool can be easily discovered by potential visitors, while ensuring a proper indexation by automatic online systems and resource recognizers. Understanding how software is depicted in scientific manuscripts and comparing these texts with the corresponding online descriptions can help to improve the visibility of the target website. It is particularly relevant to be able to align online descriptions and those found in literature, and use the resulting knowledge to improve software indexing and grouping.

Therefore, this paper presents a novel method for formally defining and mining software-related websites and related literature with the ultimate aim of improving the global online visibility of the software. As a proof of concept, the method was used to evaluate the online visibility of biomedical text mining tools. These tools have evolved considerably in the last decades, and are gathering together a heterogeneous development community as well as various user groups. For the most part, these tools are not easily discovered via general search engines. Hence, the proposed method enabled the identification of specific issues regarding the visibility of these online contents and the discussion of some possible improvements.

1. Introduction and motivation

As the World Wide Web continues to grow in contents and users, the competition for online visibility becomes evident (Pant & Pant, 2018). Online visibility is associated with brand and product awareness, product marketing, and customer relationship management, among others. Financial profit is an obvious motivation, but social engagement is also regarded as a precursor to influence users and thus, attracts the attention of both profit and non-profit organisations.

* Corresponding author at: Department of Computer Science, University of Vigo, ESEI: Escuela Superior de Ingeniería Informática. Edificio Politécnico. Campus Universitario As Lagoas s/n, 32004, Ourense, Spain.
E-mail addresses: gaeperez@uvigo.es (G. Pérez-Rodríguez), martiperez@uvigo.es (M. Pérez-Pérez), riverola@uvigo.es (F. Fdez-Riverola), analia@uvigo.es (A. Lourenço).

https://doi.org/10.1016/j.ipm.2018.11.011
Received 29 June 2018; Received in revised form 23 November 2018; Accepted 28 November 2018

Although it may seem trivial to look for software of given features or application in the Web, in practice the task may be quite challenging. There are well-established software repositories, but depending on the application area, and the social engagement of the development community, websites and specialised literature (e.g. scientific papers in the case of academia) are still the main sources of information. This problem is somewhat more serious for software developed across multiple domain fields, such as bioinformatics or computational biology, because there are often contributions from multiple development communities, which use different terminology and have their own practices in terms of information dissemination and software deposition.

This paper addresses this challenge by proposing a novel method for formally defining and mining online software descriptions and semantically aligning those descriptions with domain-specific literature. The novelty of the method lays on the ability to provide insights on the extent to which the contents of the websites are in line with domain publications, specifically how they relate to commonly used terminology, and most notably domain concepts. While getting acquainted with a given domain, potential software users become familiar with keywords and concepts meaningful to their interests, which will be likely used in later query searches. So, the semantic alignment of online descriptions with domain publications is important to evaluate the practical significance of the online contents and enable a more comprehensive indexing. At the same time, this evaluation is relevant to understand how developers present their software, i.e. features, methods, and applications, and thus, to be able to profile software communities and support software recommendation.

The field of Biomedical Text Mining (BioTM) was chosen to develop a proof of concept of the practical applicability of the method. This field has evolved significantly in the last decades and nowadays the development and user communities are quite heterogeneous. To date, there are no studies describing the online visibility of BioTM tools and resources, and the few existing software compilations are outdated and incomplete. In fact, it is quite challenging to search for specific software or keep up-to-date with the latest developments. So, the proposed method held considerable potential to uncover novel and relevant knowledge on how to enhance the online visibility of such software.

The remainder of this paper is organised as follows: a literature survey is provided in Section 2, and the context of BioTM is introduced in Section 3. Section 4 presents the proposed method, including the presentation of the multi-layer network representation and a first analysis of the website collection for the case study. The results obtained for BioTM are discussed in Section 5, namely the reconstructed multi-layer network, the clustering of software tools based on website and literature contents, the alignment of the different contents, and the insights acquired about the online visibility of the software. Section 6 summarises the work and discusses possibilities for further research.

2. Related work

The ability to evaluate the online visibility of domain-specific software and to compare online software descriptions with the corresponding (or related) publications is of practical relevance to website indexing and recommendation, namely domain-specific or vertical engines (Kassing, Oosterman, Bozzon, & Houben, 2015; Ms, Kumar, & Mukesh, n.d.). In essence, the problem falls into the research scope of semantic information retrieval, which has been boosting the emergence of alternative search strategies based on domain-specific semantics to improve the accuracy of the retrieved results (Adamo, Attivissimo, Di Nisio, & Spadavecchia, 2015; Gopalakrishnan, Sengottuvelan, Bharathi, & Lokeshkumar, 2018), as well as to determine the similarity between retrieved information (Jiang, Bai, Zhang, & Hu, 2017; Yan, Liu, Wang, Zhang, & Zheng, 2017; Zhang, Wang, Wang, Bi, & Chen, 2014).

Focusing on the target domain of application, the work of Harrow et al reviewed approaches to integrate life science literature and data sources with emphasis on semantic web, and proposed a new method for using already available open semantic web standards and technologies to integrate public and proprietary data resources, which span structured and unstructured content (Harrow et al., 2013). From a different perspective, Amer presented a novel method for enhancing the ranking performance of web search engines using ontology learning from unstructured information sources (Amer, 2015). In addition, Zhang et al developed an online semantics-based method using text features and semantic similarities to cluster Chinese web search results that outperformed the suffix tree clustering method and other traditional clustering methods (Zhang et al., 2014). In this line, a previous work also proposed a link-bridged topic model that was able to improve the prediction accuracy of cross-domain document classification based on the assumption that the documents of source and target domains share some common topics (Yang, Gao, Tan, & Wong, 2013). Finally, the work of H.-C. Yang et al presented an automatic method to generate and align multilingual hierarchies based on two types of similarity measurements, namely semantic similarity and structural similarity (Yang, Hsiao, & Lee, 2011).

3. The case of biomedical text mining

BioTM has evolved significantly in the last two decades, embracing ever more challenging industry and academia problems. The development of specialised methods and tools that enable the systematic and large-scale integration of scientific literature, biological databases and experimental data is a contemporary, well-recognised challenge of Bioinformatics (Fluck & Hofmann-Apitius, 2014; Kaushik, Baloni, & Midha, 2018; Lamurias & Couto, 2018). These tools have the potential of considerably reducing the time of database curation and enabling on-demand and highly specialised access to literature and database contents (Fleuren & Alkema, 2015; Singhal et al., 2016; Zeng, Shi, Wu, & Hong, 2015).

Community-oriented initiatives, such as BioCreative (Krallinger Martin & Valencia Alfonso, 2017), BioNLP (Cohen, Demner-Fushman, Ananiadou, & Tsujii, 2017) and BLAH3 ("Biomedical Linked Annotation Hackathon 3," 2017), are investing considerable efforts in gaining a deeper understanding about the main challenges of biomedical literature extraction and promoting the development of solutions to problems of practical interest. Most notably, these initiatives are making readily accessible annotated

literature corpora (Islamaj Dogan et al., 2017; Krallinger et al., 2015b) and enable the controlled comparison of automatic prediction systems (Krallinger et al., 2015a; Wang et al., 2016).

This is a very heterogeneous community, which is at the crossroad of Data Mining (DM), Natural Language Processing (NLP), Linguistics, and Life Sciences, among other domains, and therefore, has inherited different development practices. Currently, it is hard to keep track of existing BioTM tools and resources as well as their evolving or usage. Although some software catalogues exist, such as the BeCalm NER catalogue (Pérez-Pérez, 2017), the BioCreative software list (Krallinger, 2006) or the Omic Tools archive (Biomedical Text Mining Software Tools and Databases, 2018), scientific papers and websites are still the primary sources of information. PubMed (McEntyre & Lipman, 2001) indexes most of these publications and Google (or similar engines) enable general software search, but neither (nor in combination) are able to capture the specifics of BioTM. So, it is often difficult to search for tools or corpora based on given technical specifics (e.g. the use of a particular algorithm or technique) or biomedical applications (e.g. extraction of drug-drug interactions or recognition of gene mentions).

4. Methodology

This section starts by formally defining the problem. Consider that D designates the domain under study, S is the collection of target websites, A is the collection of accessible articles, and C is the collection of compiled concepts.

Definition 1. (domain). A domain, D, is represented by a set of concepts:

$$D = \{c_0, c_1, \cdots c_{T-1}\} \tag{1}$$

where $c_i$is the i-th concept associated with domain D, and T is the total number of concepts related to domain D.

Definition 2. (concept). A concept for a given domain, c, is represented by a set of terms (n-grams) of similar meaning:

$$c = \{t_0, t_1, \cdots t_{M-1}\} \tag{2}$$

where $t_i$is the i-th term associated with concept c, and M is the total number of terms associated with concept $c \in D$.

Definition 3. (website description). Consider that a website, s, is composed by a set of concepts, and the website description, $\vec{s}$, is composed by the distribution of all mentioned concepts and can be represented as a vector:

$$\vec{s} = \langle c_{0w}, c_{1w}, \cdots c_{N-1w} \rangle \tag{3}$$

where c_{iw} is the importance of the concept in the website description, and N is the total number of concepts mentioned in the website.

Definition 4. (article description). Consider that an article, a, is composed by a set of concepts, and the article description, $\vec{a}$, is given by the distribution of all the concepts being mentioned, and can be represented as a vector:

$$\vec{a} = \langle c_{0w}, c_{1w}, \cdots c_{N'-1w} \rangle \tag{4}$$

where c_{iw} is the importance of the concept in the article description, and N' is the total number of concepts mentioned in the article.

Definition 5. (controlled vocabulary). A controlled vocabulary, cv, can be represented as a quintuple:

$$cv = \langle O, A, C, R_{OC}, R_{AC} \rangle \tag{5}$$

where O, A, C represent infinite sets of objects, attributes and concepts, respectively. The relation R_{OC}: $O \times C \rightarrow [0, 1]$ maps the set of objects, O, to the set of domain concepts, C, for all $o_i \in O, c_i \in C$. The relation R_{AC}: $A \times C \rightarrow [0, 1]$ defines the mapping between the set of domain concepts, C, and the set of attributes (which are composed by terms, as well as other additional information like properties, descriptions or features), A, applied to describe these concepts.

Definition 6. (ontology). An ontology, Ont, can be described as a hierarchical tree defined by a septuple:

$$Ont = \langle O, A, C, R_{OC}, R_{AC}, R_{CC}^{CAS}, R_{CC}^{NCAS} \rangle \tag{6}$$

where O, A, C, R_{OC} and R_{AC} maintain the same meaning as in Definition 5, the relation R_{CC}^{CAS}: $C \times C \rightarrow [0, 1]$ maps the finite set of domain concepts, C, through the causal relations, and the relation R_{CC}^{NCAS}: $C \times C \rightarrow [0, 1]$ defines the associations (i.e. non-causal relations) among the finite set of concepts, C.

4.1. Storing and indexing structure: multi-layer knowledge network

As illustrated in Fig. 1, the proposed method organises the information in a four-layer network, which includes: the Website layer of publicly available tools; the Website concept layer generated from the contents of the websites included in the Website layer; the Article concept layer, similar to the previous Website concept layer, but derived from the Article layer; and, the Article layer, which represents existing domain literature.

Initially, all concepts belonging to the collections of websites and articles (S and A respectively) are related by association:

$$r_{c_i, c_j}^{C} = P_{c_i,cj}/P \tag{7}$$

where r_{c_i, c_j} stands for the association between the concepts c_i and c_j, $P_{c_i,cj}$ is the number of documents (i.e. websites or articles)

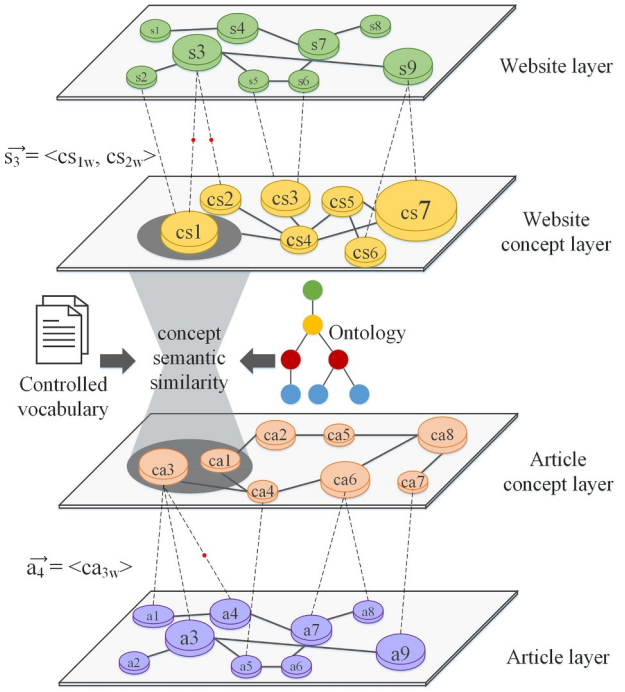

Fig. 1. Four-layer knowledge network representation, comprising the websites (green nodes in top layer), the concepts present in websites (yellow nodes in middle-top layer), the concepts extracted from articles (orange nodes in middle-bottom layer), and the articles (blue nodes in bottom layer). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

containing concepts c_i and c_j, and P is the total number of available documents. Such association represents a type of weak semantic relationship between concepts, which implies the possibility of two concepts co-occurring in a given set of documents (Xuan, Luo, Zhang, Lu, & Xu, 2016). Therefore, Website and Article concept layers are generated by connecting the existing concepts by means of Association Link Networks (i.e. ALN_{C_S} and ALN_{C_A} for websites and articles, respectively):

$$ALN_{C_S} = \begin{bmatrix} r_{0,0}^{C_s} r_{0,1}^{C_s} r_{0,2}^{C_s} \cdots r_{0,C_s-1}^{C_s} \\ r_{1,0}^{C_s} r_{1,1}^{C_s} r_{1,2}^{C_s} \cdots r_{0,C_s-1}^{C_s} \\ \vdots \vdots \vdots \ddots \vdots \\ r_{C_s-1,0}^{C_s} r_{C_s-1,1}^{C_s} r_{C_s-1,2}^{C_s} \cdots r_{C_s-1,C_s-1}^{C_s} \end{bmatrix} \tag{8}$$

$$ALN_{C_A} = \begin{bmatrix} r_{0,0}^{C_a} r_{0,1}^{C_a} r_{0,2}^{C_a} \cdots r_{0,C_a-1}^{C_a} \\ r_{1,0}^{C_a} r_{1,1}^{C_a} r_{1,2}^{C_a} \cdots r_{0,C_a-1}^{C_a} \\ \vdots \vdots \vdots \ddots \vdots \\ r_{C_a-1,0}^{C_a} r_{C_a-1,1}^{C_a} r_{C_a-1,2}^{C_a} \cdots r_{C_a-1,C_a-1}^{C_a} \end{bmatrix} \tag{9}$$

Based on the concept layer and the mapping relations between concepts and websites, the relation between two given websites, $r_{s_i,\,s_j}^{S}$, can be defined as:

$$r_{s_i,\,s_j}^{S} = \sum_{c_m \in s_i,\; c_n \in s_j} r_{c_m,c_n}^{C} \tag{10}$$

where c_m and c_n represent concepts mentioned in the websites s_i and s_j, respectively. The Association Link Network of the Website layer, ALN_S, represents all existing relations as follows:

$$ALN_S = \begin{bmatrix} r_{0,0}^{S} & r_{0,1}^{S} & r_{0,2}^{S} & \cdots & r_{0,S-1}^{S} \\ r_{1,0}^{S} & r_{1,1}^{S} & r_{1,2}^{S} & \cdots & r_{0,S-1}^{S} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r_{S-1,0}^{S} & r_{S-1,1}^{S} & r_{S-1,2}^{S} & \cdots & r_{S-1,S-1}^{S} \end{bmatrix} \tag{11}$$

Similarly, relations between two given articles, $r_{a_i,\,a_j}^{A}$, can be defined as follows:

$$r^{A}_{a_i,\,a_j} = \sum_{c_m \in a_i,\ c_n \in a_j} r^{C}_{c_m,c_n} \tag{12}$$

where c_m and c_n represent the concepts mentioned in the articles a_i and a_j, respectively. So, the Association Link Network of the Article layer, ALN_A, represents all existing relations as follows:

$$ALN_A = \begin{bmatrix} r^{A}_{0,0} & r^{A}_{0,1} & r^{A}_{0,2} & \cdots & r^{A}_{0,A-1} \\ r^{A}_{1,0} & r^{A}_{1,1} & r^{A}_{1,2} & \cdots & r^{A}_{0,A-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ r^{A}_{A-1,0} & r^{A}_{A-1,1} & r^{A}_{A-1,2} & \cdots & r^{A}_{A-1,A-1} \end{bmatrix} \tag{13}$$

4.2. Populating the network: website and article layers

Given a set of target websites and the corresponding collection of background knowledge (i.e. articles from the literature), the population of the four-layer network representation starts by generating Website and Article concept layers, i.e. meaningful concepts are identified, concept weights (c_{iw}) are computed based on term frequency-inverse document frequency (tf-idf) values (Ramos, 2003), and the co-occurrence associations among concepts are established. Then, the mapping relations between concepts and websites or articles are applied, and the similarity between websites and articles, which are represented as concept vectors ($\vec{s}$ and $\vec{a}$, respectively), are computed using the cosine method (Salton & Buckley, 1988). Based on these mapping relations, Website and Article layers (i.e. ALN_S and ALN_A, respectively) are then constructed.

The Website and Article layers can be compared through their concept layers. That is, concept layers can be mapped to each other by applying a semantic query expansion, which includes a mathematical model to compute semantic similarity between article and website concepts, and an algorithm for query expansion based on some ontology or controlled vocabulary that describes the domain terminology (the availability of such semantic resources varies from domain to domain) (Chien, Liu, Wu, Lai, & Huang, 2016; Fernández-Reyes, Hermosillo-Valadez, & Montes-y-Gómez, 2018; Yunzhi, Huijuan, Shapiro, Travillian, & Lanjuan, 2016).

Despite the specificities of the domain, it is possible to establish a common workflow to retrieve and process the information that will feed the multi-layered knowledge network. Fig. 2 presents the general workflow proposed to retrieve, process and analyse the information used to populate the Website layer. A web crawler supports the automatic retrieval of website contents. Then, several NLP (e.g. tokenisation, Porter stemming or n-gram processing) (Bernstein, 2018) and text mining (TM) (e.g. entity recognition) techniques are to be applied to the website contents to extract concepts deemed of practical interest to the analysis of the software domain. The use of controlled vocabulary or ontologies of domain relevance may be quiet handy at this point. The recognised concepts are applied to the population of the Website concept layer.

Simultaneously, the Article network layer may be initialised by querying a public bibliographic repository or using a general search engine. The mechanisms used to retrieve those articles depend on the programmatic access methods available for the engine. As minimum information requirements, an article should be described by the bibliographic identifier, title, journal, authors, publication date, and abstract. Then, and following the same strategy as in the Website concept layer, text processing techniques and entity recognisers should be applied to identify meaningful concepts and populate the Article concept layer.

4.3. Reasoning: data mining and network analysis through concept layers

Depending on the availability of an appropriated ontology representing the domain of knowledge, there are two alternatives to calculate the semantic similarity of the concepts (Fig. 1). The recommended, general approach to such calculation takes into account metrics such as the distance between nodes (i.e. concepts), the depth and the node density, as proposed earlier by Yunzhi et al. (2016).

Whenever an ontology is available, distance refers to the number of edges composing the shortest path between two nodes (n_i and n_j) in the ontology structure. Therefore, the distance between two concepts, c_i and c_j, can be expressed as:

$$dist\,(c_i,\, c_j) = e^{-dist\,(n_i,nj)} \tag{14}$$

that is, the greater the distance between two concepts, the lower is the similarity. When the distance of two concepts is zero then their similarity is 1, i.e. it is the same concept. Likewise, when the distance between two concepts tends to infinity then the similarity tends to 0, i.e. the concepts hold no semantic relation.

The depth between two concepts, c_i and c_j, can be expressed as the depth relationship between the corresponding nodes in the ontology:

$$depth\,(c_i,\, c_j) = \frac{|depth\,(c_i) - depth\,(c_j)| + 1}{depth\,(c_i) + depth\,(c_j)} \tag{15}$$

where the depth of a given concept, $depth(c_i)$, represents its shortest path to the root (i.e. $depth\ (c_i) = sp + 1$, where sp is the shortest path between c_i and the root node). The depth of the root is always 1. The smaller the depth of two concepts is, the greater the similarity between concepts is.

Finally, the density between two concepts, c_i and c_j, is defined as the child node density of the common ancestor node between the

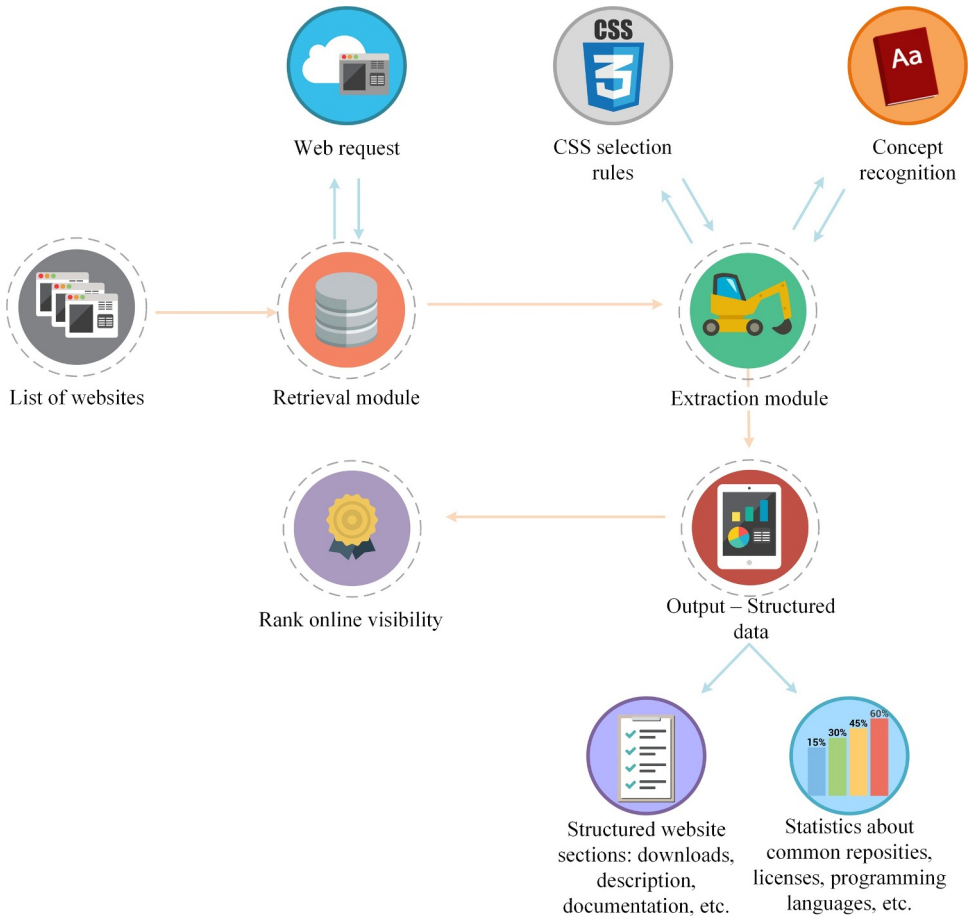

Fig. 2. General workflow for the population of the Website layer, including the retrieval of the list of target websites, the extraction of the corresponding contents, and the semantic, domain-specific analysis of these contents.

two nodes, and is expressed as follows:

$$density\ (c_i, c_j)\ = n/m \tag{16}$$

where n is the number of direct child nodes of the common ancestor node of c_i and c_j, and m is the number of all child nodes of the common ancestor node of c_i and c_j. That is, the density of the child node is directly proportional to the refinement of the node. Greater density implies a more refined concept.

Taking into account all of the above metrics, the semantic similarity of two given concepts, c_i and c_j, is calculated using the following expression:

$$sim(c_i, c_j) = \ \alpha \times dist(c_i, c_j) + \beta \times depth(c_i, c_j) + \lambda\ \times density(c_i, c_j) \tag{17}$$

where α, β, λ are adjustment factors that weight the relative importance of each metric, being $\alpha + \ \beta + \ \lambda = 1$.

In case a suitable ontology is not available, the semantic similarity of the controlled vocabulary used may be simplified to:

$$sim(c_i, c_j) = \ dist(n_i, n_j) \tag{18}$$

where the depth and density of the two concepts are 0, and the similarity between them is 1 if they represent the same concept or 0, otherwise.

The above described calculation is exemplified in Fig. 3, which presents a hypothetical ontology tree, where the different levels of the structure are denoted by colour.

Eqs. (19)–(21) illustrate the calculation of the values of distance, depth and density for the nodes c_1 and c_7, respectively. The semantic similarity of nodes c_1 and c_7 is finally calculated in Eq. (22).

$$dist(c_1, c_7) = e^{-2} = 0.13 \tag{19}$$

$$depth(c_1, c_7) = \frac{|1 - 3| + 1}{1 + 3} = 0.75 \tag{20}$$

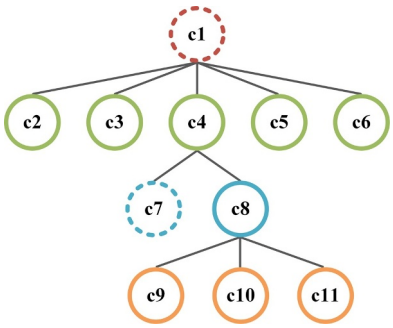

Fig. 3. Example of a hypothetical ontology tree for similarity computation. (For interpretation of the references to colour in the text, the reader is referred to the web version of this article.)

$$density(c_1, c_7) = \frac{5}{10} = 0.5 \tag{21}$$

$$sim(c_1, c_7) = \underbrace{\alpha}_{0.6} \times\ 0.13 + \underbrace{\beta}_{0.2} \times\ 0.75 + \underbrace{\lambda}_{0.2} \times\ 0.5 = 0.32 \tag{22}$$

The calculation of the semantic similarity of the concepts under analysis enables the application of data mining methods, which can be straightforwardly applied for the systematic analysis of the Website and Article concept layers. The primary goal of such analysis is to identify the terminology being used to describe existing software, namely general software terminology, domain-specific vocabulary, and application-specific language. Most notably, those terms selected by the developers as the most representative of the software being publicised in the web, and the terminology chosen by both developers and users to describe software in the literature. To this end, clustering methods can aid to identify software communities, defined and related by terminology, from the contents of websites and articles. Moreover, by aligning the two sets of clustering results, it is possible to identify discrepancies between online and article descriptions, and observe how these affect software similarity assessment, and implicitly, further indexing and recommendation strategies.

4.4. Quantifying the online visibility score

When evaluating online visibility, it is important to be able to measure the quality of the website representing the target tool or resource. Such rank indicator is directly related with (*i*) the similarity between the website description ($\vec{s}$) and the article description ($\vec{a}$), (*ii*) the technical contents available in the website, and (*iii*) the suitability of the website description ($\vec{s}$) when compared with the analysed domain. Considering these factors, Eq. (23) makes it is possible to compute a single summary measure of the online visibility as follows:

$$Online\ visibility\ =\ \alpha^{*}Sim\ (\vec{s},\ \vec{a}) +\ \beta^{*}\ Tech\ (\vec{s}) + \gamma^{*}Dom\ (\vec{s}) \tag{23}$$

where α, β, γ are adjustment factors that weight the relative importance of each metric, being $\alpha + \beta + \gamma = 1$.

$Sim\ (\vec{s},\ \vec{a})$ expresses the cosine similarity between the contents of the website and the corresponding article description. That is, this metric is helpful to understand how difficult can be to find tools in the web when using the terminology found in the associated literature.

$Tech\ (\vec{s})$ models the visibility of basic information that should be present in the website and be readable for both, humans and machines. Information such as the tool name, license, supported operating systems or software repository are of common interest to most users. Arguably, it is important to avoid a high fragmentation of the documentation using multiple hyperlinks because it worsens the usability of the website. When the available documentation lacks clarity or enough detail, it is important to have a way of contact to make questions, in addition of giving the possibility of using social networks. The calculation of $Tech\ (\vec{s})$ reflects all these considerations as defined by Eq. (24):

$$Tech(\vec{s}) = \frac{Tech' + H.\ ReadableName}{4} \tag{24}$$

where *H.ReadableName* indicates the presence of the name of the tool, being equal to 1 if it could be recognised and 0 otherwise, and *Tech′* is computed using Eq. (25):

$$\boldsymbol{Tech}' = \left(\frac{\boldsymbol{H.License}}{\boldsymbol{2 - I.OSI}} + \boldsymbol{H.\ OS} + \boldsymbol{H.\ Repository} \over 3 \right) + \frac{\boldsymbol{H.\ Documentation}}{\boldsymbol{1 + \sum\ Links \in Documentation}} + \frac{\boldsymbol{H.\ Contact}}{\boldsymbol{2 - H.\ SocialMedia}} \tag{25}$$

where the prefix *H.* denotes existence (“Has”), being equal to 1 if the website contains the attribute and 0 otherwise, the prefix *I.* denotes definition (“Is”), being equal to 1 if the attribute satisfies the condition and 0 otherwise, and OSI denotes Open Source Initiative license.

Finally, $Dom\ (\vec{s})$ represents the suitability of the information provided by the website. The index page should have a minimum of information about the project scope and the proposed tool, notably main technologies and features relative to the proposed domain. Eq. (26) describes how this metric is computed.

$$Dom\ (\vec{s}) = \frac{\sum_{c \in Index}}{\sum_{c \in Website}} \times \frac{1}{1 - \left(e - e^{1 - \left(\frac{1}{\sum_{c \in Website} + 2} \right)} \right)} \tag{26}$$

where the summation in the numerator represents the number of concepts present in the index page of the website, whereas the summation in the denominator stands for the number of concepts present in the whole website, and e is the Euler's constant.

5. Proof of concept: BioTM tools

The practical applicability of the proposed approach was evaluated over two real-world datasets representing websites and scientific articles describing publicly available BioTM software. The main goal was to measure the online visibility of BioTM tools and to compare the contents of websites and scientific articles describing the development or the application of such software.

The list of articles to be analysed was initially defined based on searches over PubMed (McEntyre & Lipman, 2001), which holds one of the most comprehensive literature catalogues for Life Sciences (Biomedical Text Mining software tools and databases, 2018). Queries were customised to retrieve articles presenting or reporting the use of BioTM tools and corpora. In particular, the article abstracts should mention terms related to TM or NLP, and the use or presentation of software in any common 'form' (e.g. mentions to a tool, workbench, corpus or application). The National Center for Biotechnology Information (NCBI) Entrez Utilities Web services (Entrez Utilities Web services, 2010) were used to programmatically access the PubMed library and download article information to be further processed. The resulting dataset contained 5,764 unique BioTM-related articles that were used to populate the Article layer depicted at the bottom in Fig. 1.

The list of target websites was obtained from those articles containing URLs (i.e. 'http/s' alike strings). Considering the extensive manual curation required by this step, the search for websites was wide but not comprehensive. The aim was to have an unbiased and general view of existing online contents. Initially, 324 websites were identified as possibly interesting. This list was then curated in order to eliminate broken links and URLs related to software repositories (e.g. GitHub or SourceForge). Additionally, some software catalogues (Krallinger, 2006; Pérez-Pérez, 2017) were screened to enhance the diversity of the dataset, namely to include tools that were not being explicitly mentioned in the collection of articles (e.g. proprietary software). As a result, a final list of 135 unique BioTM websites was used to populate the Website layer represented at the top in Fig. 1. The two datasets are described in Supplementary Material 1.

The JSOUP java API (Hedley, 2017) was applied to perform website crawling, ensuring compliance with common crawling policies and practices, namely *robots.txt* instructions. To avoid unnecessary requests, website contents were stored in a local cache and the crawler was instructed to ignore several unnecessary website resources (e.g. compressed files, images and videos).

Similar text processing was applied to website contents and article abstracts. In particular, text tokenization, stemming, and stop word removal were performed using the tools of the Apache Lucene Core (Bernstein, 2018). The JSOUP HTML parser and CSS debugger, domain-specific dictionaries, and a customised set of regular expressions enabled the recognition of meaningful terms related to general software concepts (e.g. programming languages, operating systems and licenses), domain-specific terms (e.g. NLP methods and DM algorithms), and application-specific concepts (e.g. gene recognition and extraction of drug-gene interactions).

DM experiments were conducted in RapidMiner (Hofmann & Klinkenberg, 2013) and network analyses were performed using Cytoscape v3.5.1 (Shannon et al., 2003).

5.1. Preliminary analysis of website contents

Following the general workflow proposed in Fig. 2, this section introduces the preliminary analysis of the BioTM website contents, which highlights how different these descriptions are from a more technical point of view. In particular, Fig. 4 shows the number of websites that make reference to operating systems, programing languages, licenses, version control repositories and social media support services (bearing in mind that a given website can hold multiple values for a given category, e.g. a tool can run in multiple operating systems).

There is also information about the existence of metadata description tags and available documentation. Noteworthy, the metadata description tag, which is a key indexing element for common search engines, was only found in 33 websites. In the absence of this information, search engines typically resort to whole contents, which is a challenging processing task (namely due to website structure) and may lead to inconsistent/unintended results.

Regarding other technical features, such as programming languages, documentation, operating systems or social media, the number of mentions was, in general, somewhat low (mentions found in less than 67% of the websites). For those websites that did contain such mentions, the most mentioned programming language was Java, followed at a distance by Python. Likewise, the most mentioned software repositories were SourceForge and GitHub. References to operating systems or license were found in less than 35% of the websites, and documentation (in different forms, i.e. pages labelled as documentation, tutorial, help, FAQ, quick start or API) was found in 67% of the websites, approximately. Supplementary Material 2 discloses the details of this preliminary website analysis.

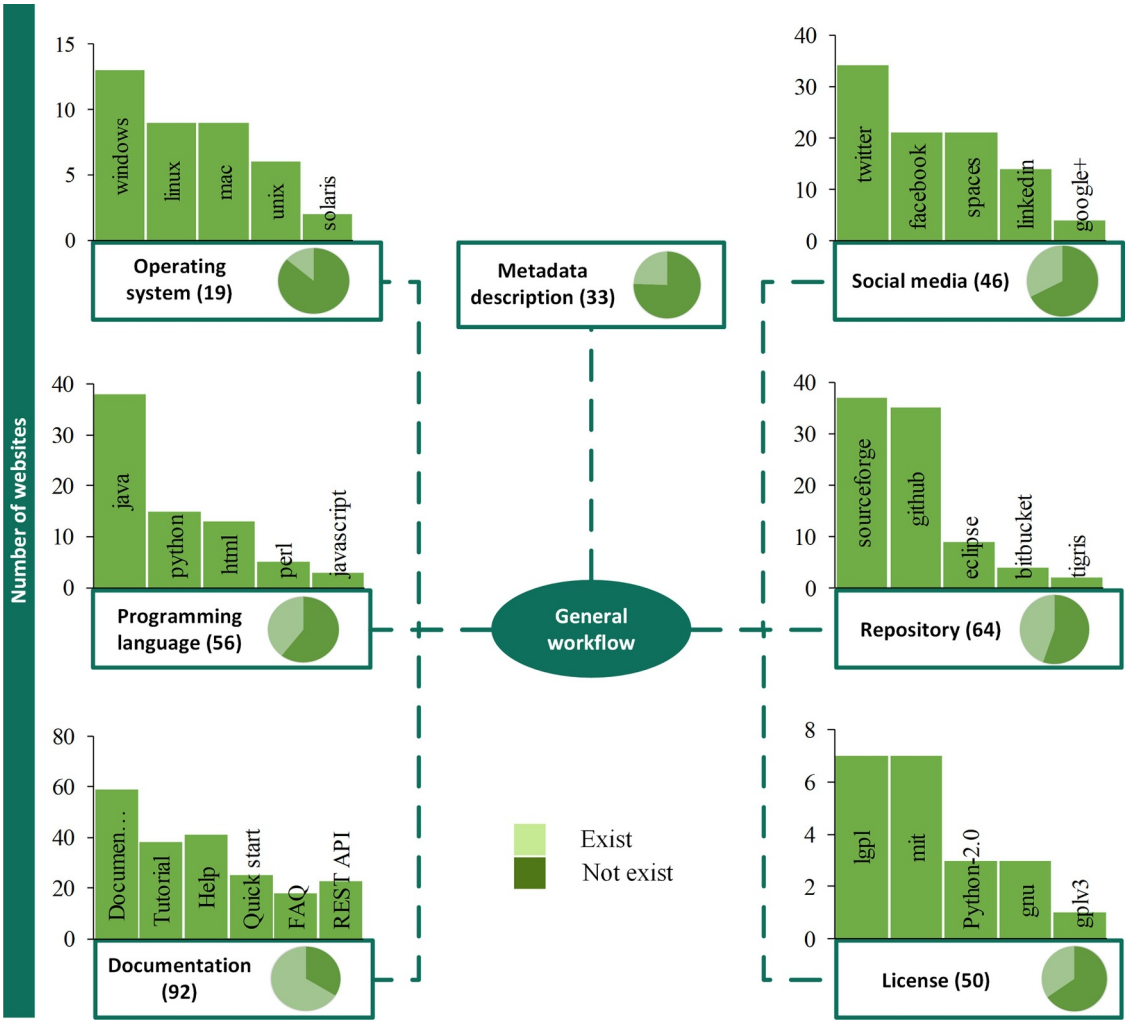

Fig. 4. Technical contents referenced in the BioTM websites analysed. The number of websites that make reference to a particular technical aspect is given in parentheses. The *Y* axis represents the number of websites including mentions to the category depicted in the *X* axis.

5.2. Multi-layer representation of available knowledge

The multi-layer knowledge network was populated with BioTM tools and articles following the procedure previously explained in Section 4.2. Fig. 5 illustrates the Website layer (top layer in Fig. 1), which describes the relations established between different BioTM tools. The nodes are coloured using a gradient from dark to light green based on the node degree; the same occurs with the node size. That is, bigger and greener nodes represent tools that have more concepts in common with the other tools, based on website contents. For example, *ABNER* shares concepts with the majority of the other BioTM tools (i.e. 122 tools), whereas *sherloc2* is only related with 57 of them. In general, the extent of concept sharing between websites can be explained in two ways: (i) some websites have more textual contents (e.g. more extensive documentation) and thus, are more likely to mention a higher number of concepts (e.g. *ABNER*), whereas other websites only have a short description and the number of concepts being mentioned is low (e.g. *sherloc2*); and, (ii) some website descriptions are more technical while others remain more general.

The same strategy is applied to the construction and interpretation of the contents of the Article layer (bottom layer in Fig. 1). This network is presented in Supplementary Material 3.

In the corresponding concept layers (middle layers in Fig. 1), nodes are coloured based on the category of the concepts (Figs. 6 and 7). In particular, the colour orange denotes BioTM resources, the colour green denotes biomedical applications, the colour yellow denotes BioTM-specific technical concepts, and the colour blue denotes general technical concepts. In the absence of a publicly available ontology describing the BioTM technical terminology, a controlled vocabulary was prepared in-house based on existing knowledge of the terminology commonly used by those working in the fields of BioTM, NLP and DM, as well as general characterisation of biomedical applications. Complementarily, the calculation of the tf-idf scores, c_{iw}, of uni-, bi- and tri-grams found in the texts (both in website and article contents) and the manual inspection of the top scoring terms, enabled the compilation of further

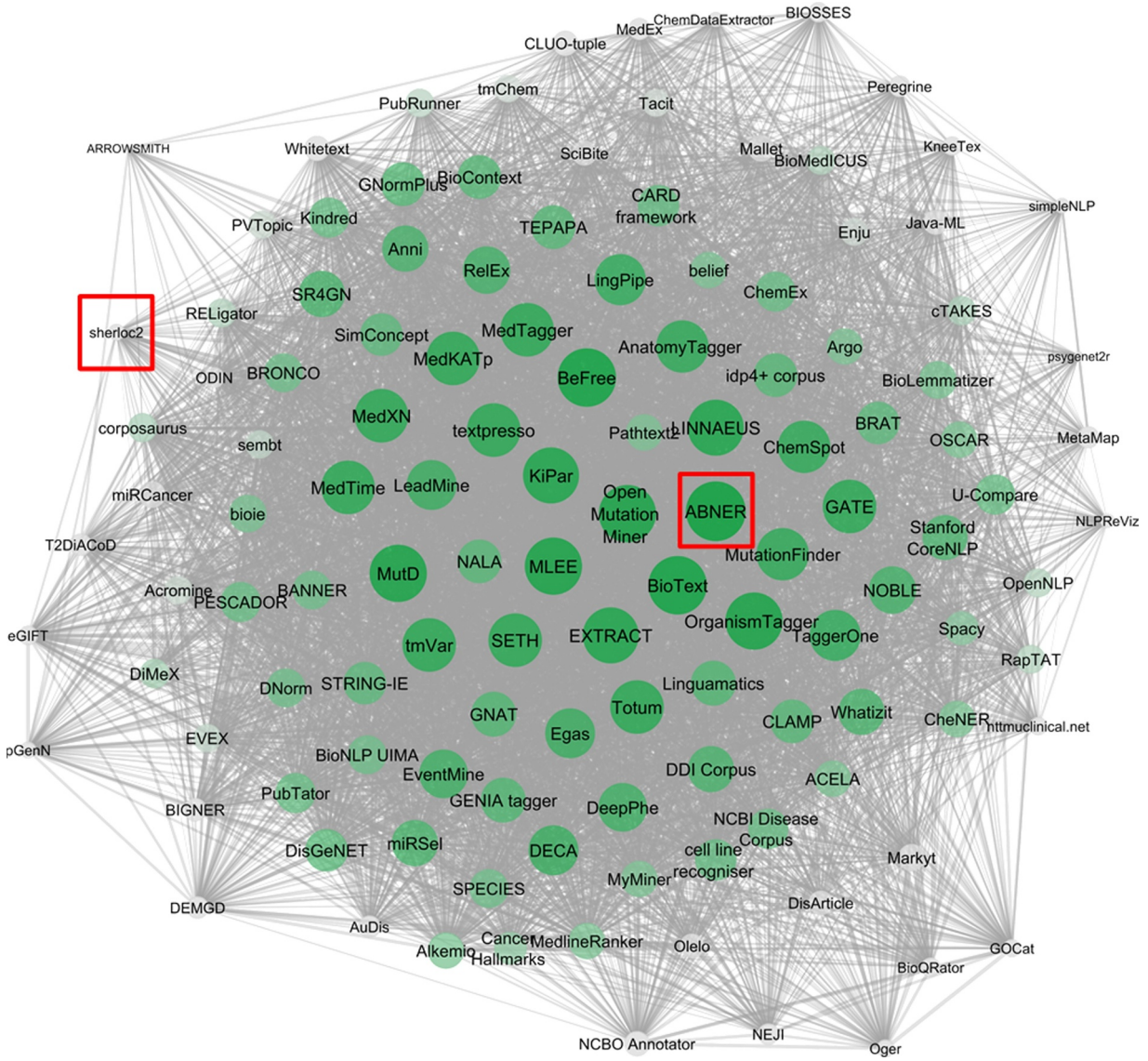

Fig. 5. Representation of the BioTM Website layer. The node size and colour are based on the node degree, whilst the edge size is based on the similarity of the corresponding website concept vectors. Two websites are related if they share, at least, one descriptive concept.

terminology (new concepts or synonyms for already included concepts).

Fig. 6 presents the resulting Website concept layer. The size of the nodes is based on the node degree (i.e. bigger nodes represent concepts that are more mentioned in the websites), the node colour relates to the concept category, and the edge size is based on the strength of node co-occurrence. For example, the concept '*corpus*' is one of the biggest nodes, being mentioned in the contents of 55 websites, whereas the concept '*CLEF*' (i.e. the name of a given corpus) is only mentioned in 2 websites. Also of notice, most of the nodes represent BioTM-specific technical concepts (80), followed at a distance by general technical concepts (49), BioTM resources (28), and biomedical applications (28). Overall, the Website concept layer shows a balance among technical concepts (specific to BioTM or more general) and concepts referring to bio-applications and resources. The biggest node for each category are '*corpus*', '*java*', '*gene*' and '*AIMed*', respectively. These data meet the common belief that most developers highlight main technical features and potential applications when describing their products.

Complementarily, Fig. 7 presents the Article concept layer where the colour of the nodes, the node size and the edge size maintain the same semantics as described for Fig. 6. At a first sight, this collection of nodes maintains a similar trend, i.e. there is a majority of BioTM-specific technical concepts (108), followed by general technical concepts (51), concepts related to BioTM resources (36) and biomedical applications (35).

However, unlike the previous network, the BioTM-specific and bio-application concepts are more common than general technical concepts (visible in terms of the node size). For example, '*text mining*', '*corpus*', or '*protein*' are domain-specific concepts are mentioned

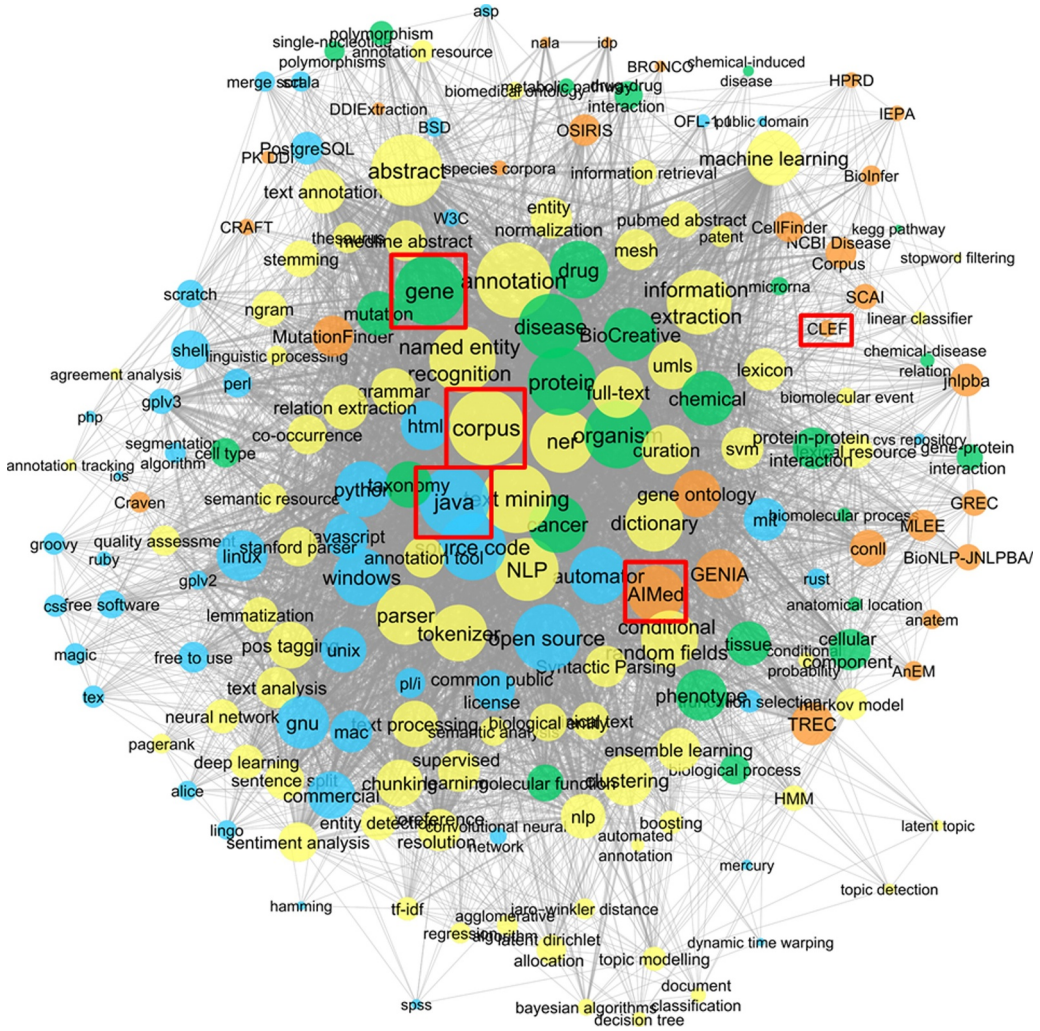

Fig. 6. Representation of the Website concept layer. The node size is based on the node degree while the node colour relates to the concept category (i.e. orange denotes BioTM resources, green denotes biomedical applications, yellow denotes BioTM-specific technical concepts, and blue denotes general technical concepts). The edge size is based on the strength of the co-occurrence among the nodes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

in many articles (1614, 773 and 753, respectively), while '*convolutional neural network*', '*lexical analysis*' or '*knn*' are among the most mentioned general technical concepts (13, 8 and 8, respectively). Regarding the resources category, the number of concepts is nearly the same as in the previous layer and, most notably, the most common concept, which is the '*AIMed*' corpus, is the same. Overall, network statistics agree with the common template of software presentation in scientific articles. In particular, the specifics of the implemented software often have a major relevance in these articles, followed by the description of a proof of concept or some case of study.

5.3. Coherence of BioTM online contents from literature point-of-view

Considering another perspective of analysis, online contents and literature descriptions can be explored in terms of clustering similarities. As illustrated in Fig. 8, tools can be represented as coloured dots, where colours denote cluster assignments. Therefore, bicoloured nodes identify software that present enough dissimilarities between online contents and literature descriptions to justify a different cluster assignment. The left hand colour represents the cluster assignment based on the website contents, whereas the right hand colour represents the cluster assignment based on the literature description. For example, the online description of the Bio-Lemmatizer is dominated by the general concepts 'biomedical' and 'annotation' (i.e. most of the web information is about installation and usage), whereas the supporting article includes a number of BioTM-related technical concepts, such as 'lemmatization' or 'pos tagging'.

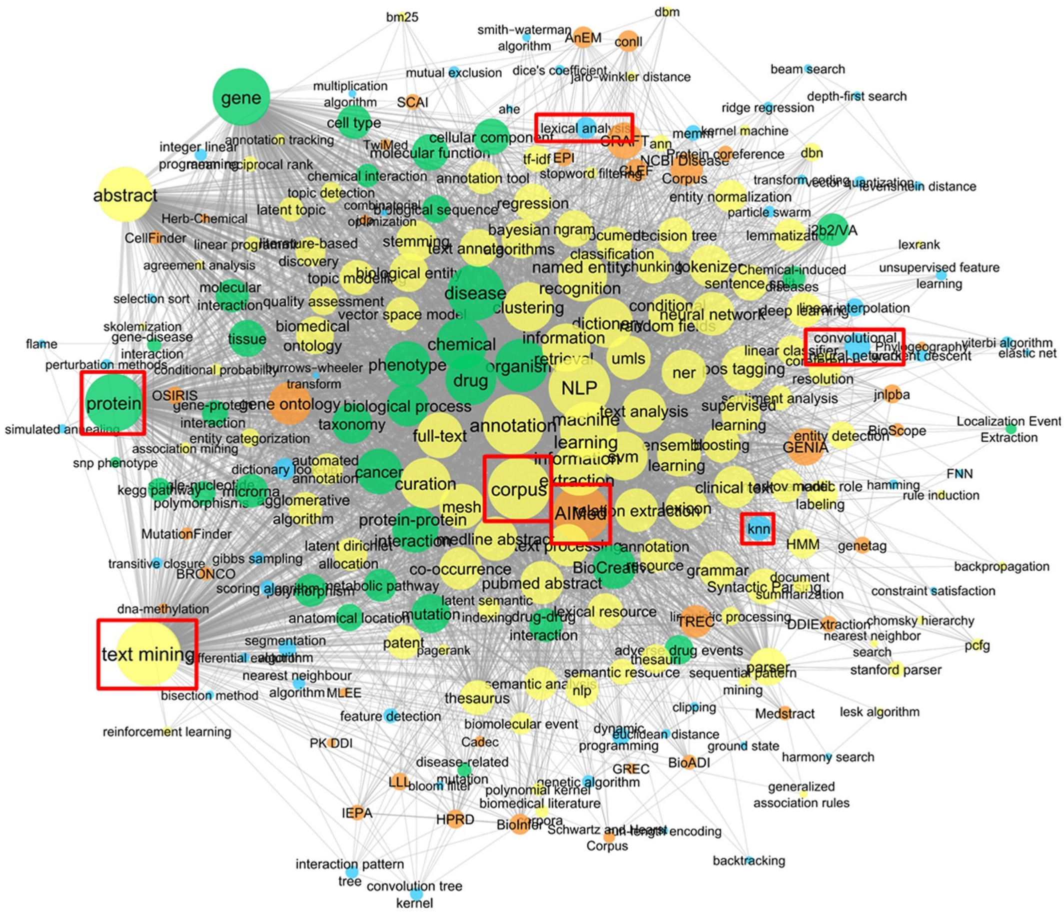

Fig. 7. Representation of the Article concept layer. The node size is based on the node degree while the node colour represents the concept category (i.e. orange denotes BioTM resources, green denotes biomedical concepts, yellow denotes BioTM-specific technical concepts, and blue denotes general technical concepts). The edge size is based on the strength of the co-occurrence among the nodes. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

These discrepancies may be an issue when developing recommendation models based on general or BioTM-related concepts. Considering BioLemmatizer again, this tool would be presented as a recommendation grouped with different tools depending on the selected concepts, i.e. if general concepts are chosen, the tool will lay in the orange cluster, otherwise the tool will lay in the blue cluster. In this context, it should be considered the alignment of the both contents to allow the use of query expansions in order to consider a greater number of relevant tools.

5.4. Identifying BioTM software communities

An additional point of interest in the present study is the clustering of BioTM software based on online contents (Fig. 9). The in-depth analysis of this model is of practical relevance to both software developers and users. From the developer's point of view, this analysis allows the identification of similar (competitor) tools as well as underdeveloped software niches. From the user's point of view, it is interesting to look into the concepts that are more popular in online contents and those that are unlikely to retrieve any candidates. Likewise, it is possible to inspect if the user may discover BioTM software indirectly through its semantic associations, i.e. while not containing concepts matching the user query, semantic similarity to other tools including such concepts enables tool retrieval.

In Fig. 9, BioTM tools are again coloured according to their cluster assignment. The distribution (position) of the software in the figure is based on the cosine distance between the corresponding concept vectors, $\overrightarrow{s_i}$ and $\overrightarrow{s_j}$, i.e. the closest the data points, the more similar are the corresponding online contents. Therefore, two BioTM tools that are close to each other, but are assigned to different clusters, are represented by similar concepts with different concept frequencies, c_{iw}.

Overall, it is possible to observe four clusters with the following incidence of concepts: the cluster coloured in green is focused on ‘*organism*’, ‘*annotation*’ and ‘*gene*’; the cluster coloured in orange is mainly represented by the concepts ‘*annotation*’, ‘*biomedical*’ and

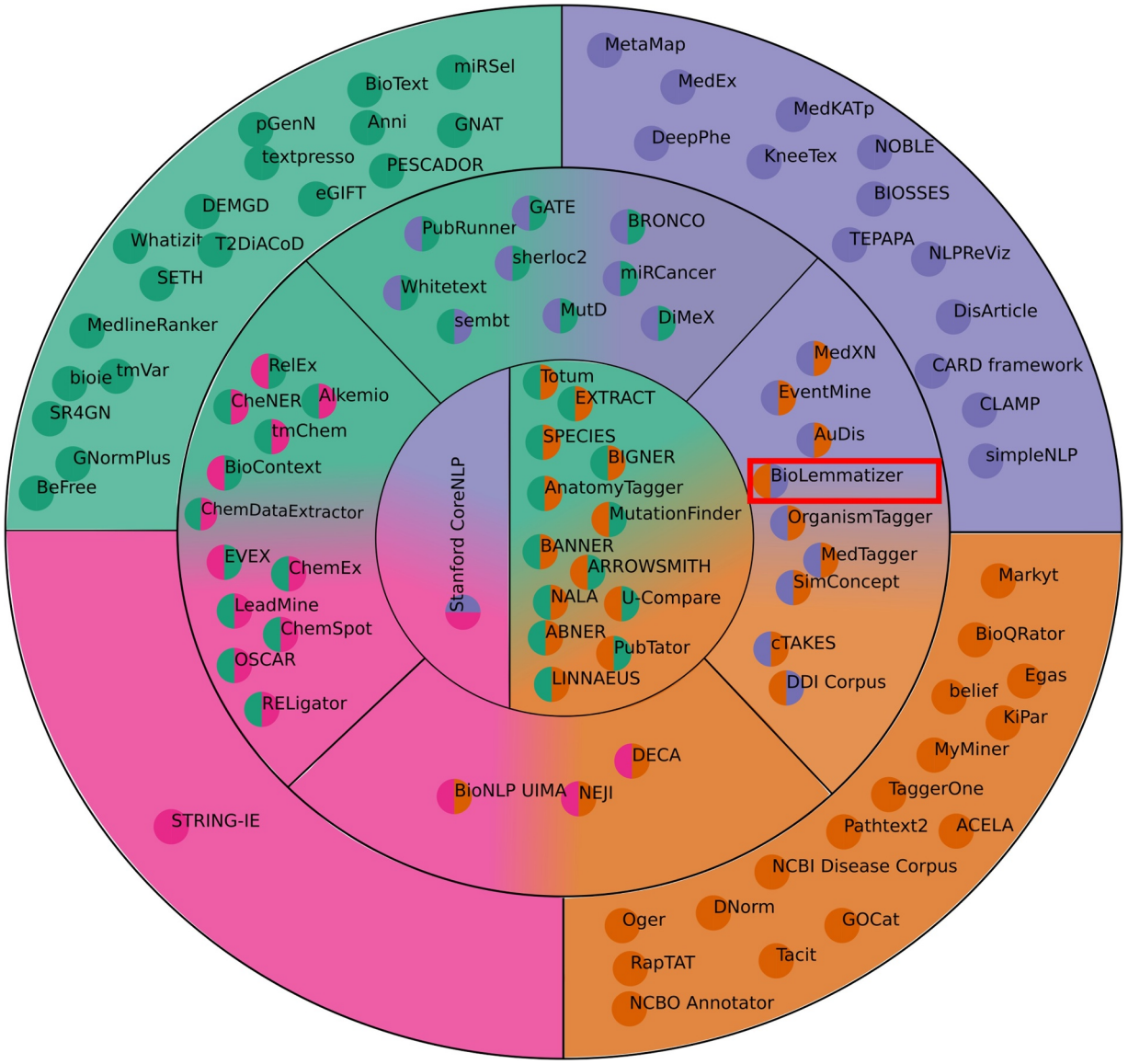

Fig. 8. Representation of existing BioTM tools based on online contents and literature descriptions. Bicoloured nodes denote tools clustered differently depending on the contents under analysis (the left hand colour represents website contents clustering, and the right hand colour stands for literature description clustering). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

'*corpora*'; the cluster coloured in purple is centred in the concepts '*biomedical*', '*NLP*' and '*cancer*'; and finally, the cluster coloured in pink has '*pos tagging*', '*parser*' and '*NLP*' as main concepts. According to the distribution of the concepts in the different clusters, it is possible to observe that, for the most part, the green cluster contains automatic annotators, the manual annotation tools fall into the orange cluster, the purple cluster has the bio-specific NLP tools, and the pink cluster contains generic NLP tools.

Looking into the placement of specific tools, one may uncover unintended and previously unknown "deficiencies" in how the tools are presented online. For example, the tools *tmVar* and *SimConcept* are close because they share concepts such as '*conditional random field, 'gene'* and '*disease'*, but they belong to different clusters because '*conditional random field*' and '*annotation*' (i.e. algorithm and task, respectively) are part of the centroid representing the green cluster whereas '*cancer'* (i.e. application) is part of the centroid representing the purple cluster. With this information in hands, developers may plan website content enhancements to potentiate or shift current online positioning.

5.5. Unravelling the importance of relevant concepts

From a complementary point of view, being able to navigate within each layer and across network layers also enables a

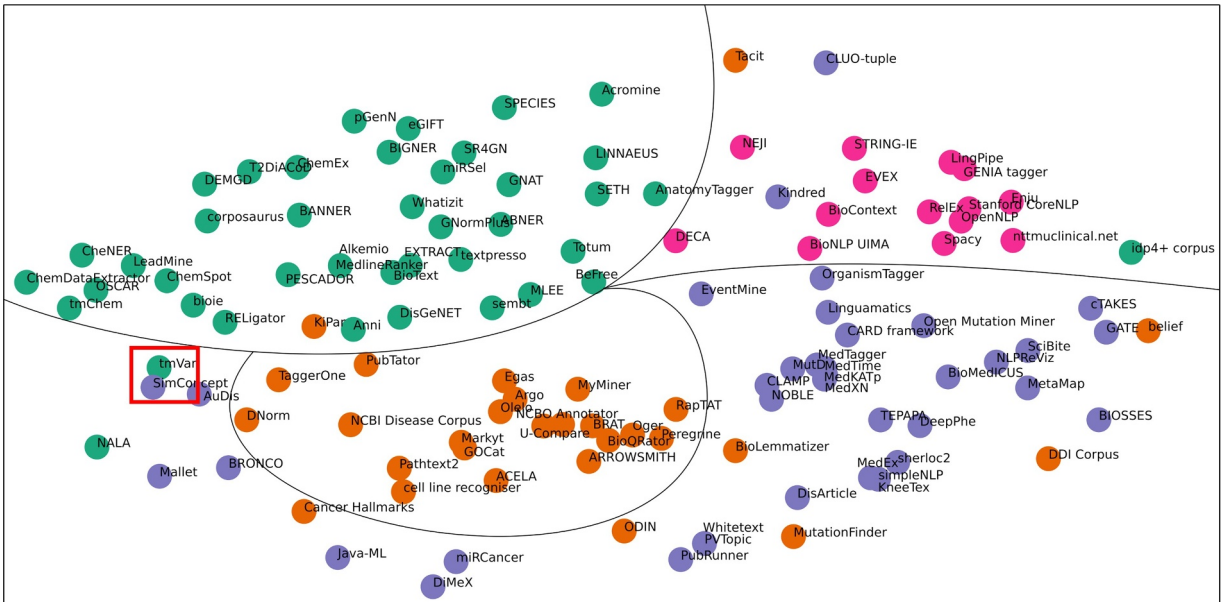

Fig. 9. Clustering of BioTM tools based on online contents. Spatial distribution is based on the multi-dimensional scaling of the cosine distance between websites. Data points are coloured based on cluster assignment. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

comprehensive analysis of the methods and techniques currently applied by different BioTM tools. Figs. 10–12 illustrate (in two-dimensional scatter plots), the distribution of the software based on the mention of given concepts in website contents. Each data point represents a BioTM tool, coloured according to cluster assignment. The *X* and *Y* axis represent the value of the weighted concepts, c_{iw}, for each website. Moreover, in order to enhance the visibility of some points, a jitter was applied to the chart and the dense areas are zoomed.

Fig. 10 shows the distribution of existing tools according to technical concepts typically used in the domain of '*NLP*' (e.g. mentions to '*thesaurus*', '*pos tagging*' or '*tokenization*') and those belonging to the domain of '*Machine learning (ML)*' (e.g. mentions to supervised learning algorithms, such as support vector machines and hidden Markov models). Considering that BioTM is at the cross of multiple Computer Science domains, this kind of analysis is interesting to observe whether developers highlight the combination of techniques from multiple domains, the use of innovative approaches, or simply point out the core area of development. For example, the online contents of *Stanford CoreNLP, Spacy* and *LingPipe* have a high occurrence of concepts from both *ML* and *NLP*, whereas the online contents of *Mallet* and *Java-ML* emphasise ML concepts.

Fig. 11 illustrates the distribution of BioTM tools considering the description of NLP techniques and methods (used in tool implementation) versus biomedical applications. For example, the concept vector describing *BeFree*, $\overrightarrow{s}_{BeFree}$, stresses concepts such as '*pos tagging*' and '*disease*' for NLP and biomedical applications, respectively. Additionally, online descriptions of those tools positioned at the right lower corner (belonging to the blue and green clusters) include a high frequency of NLP concepts but provide little information about biomedical applications. For instance, the concept vector describing *Stanford CoreNLP*, $\overrightarrow{s}_{Stanford\ CoreNLP}$, underlines concepts like 'pos tagging', 'tokenizer' or 'lemmatization', which refer to NLP techniques.

Following the same scheme, online descriptions of existing tools positioned at the left upper corner (especially those belonging to the red cluster) possess a high frequency of biomedical application concepts but lack of information about NLP. For example, the concept vector describing *SPECIES*, $\overrightarrow{s}_{SPECIES}$, manages concepts like 'taxonomy' and 'organism' for biomedical applications.

Finally, Fig. 12 portrays the software in terms of a specific TM task, i.e. named entity recognition (NER), and main biomedical applications, namely the recognition of genes, organisms, and proteins. It may be observed that the online contents of the tools *DECA, SPECIES* and *eGIFT* clearly describe the biomedical applications without much reference to the task, whereas the descriptions of *LINNAEUS, Anatomy Tagger*, and *GENIA* emphasise the NER task, but provide little information about possible biomedical applications.

5.6. Computing the online visibility of BioTM software

Aside from the semantic analysis of online contents, it is relevant to analyse these contents in terms of general readability and access. Table 1 shows the BioTM tools that obtained the highest and lowest scores (top 3). Scores were calculated using Eq. (23). Values of α, β and γ were set to one third each to maintain a similar proportion among components (i.e.

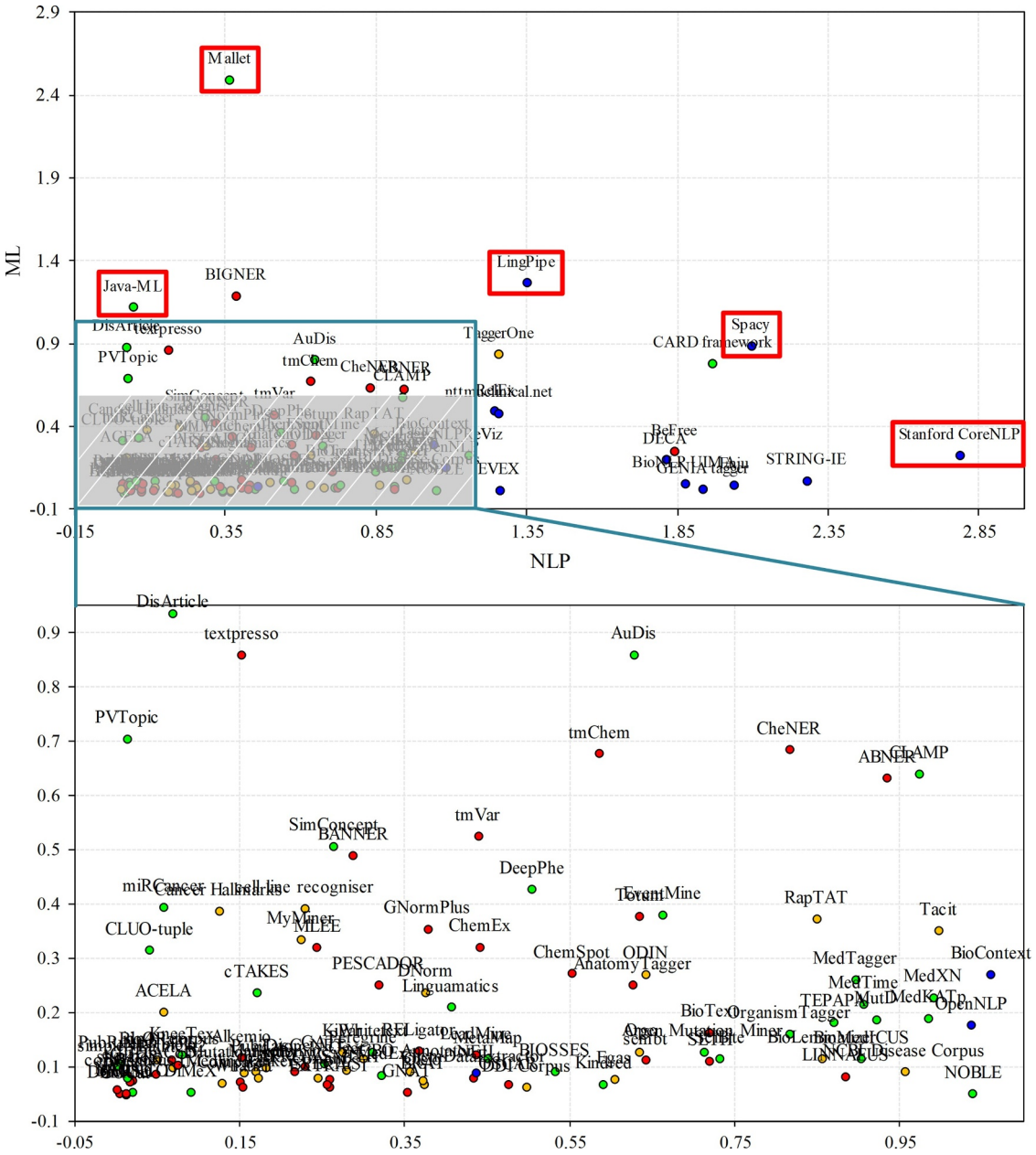

Fig. 10. Distribution of BioTM tools according to the use of terminology related to NLP (*X* axis) and ML (*Y* axis). Colour denotes cluster assignment. The shaded area highlights points with poor incidence of these concepts. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Sim ($\vec{s}$, $\vec{a}$), *Tech* ($\vec{s}$) and *Dom* ($\vec{s}$)).

OrganismTagger, DDI Corpus and *ChemSpot* were the three BioTM tools that achieved the highest scores. Noteworthy, the online contents of these tools are presented into a single HTML page. Given the similar weight associated to each component of Eq. (23), it can be observed that although *DDI Corpus* presents a low value for *Tech* ($\vec{s}$) when compared to the other tools, it is compensated by a high affinity with its associated scientific publication (*Sim* ($\vec{s}$, $\vec{a}$)). Moreover, these websites provide information about available downloads, the use of the software, and align reasonably well with the corresponding articles.

On the other hand, *MedXN, MPTM* and *MutD* were the BioTM tools that achieved the lowest values. The websites of *MedXN* and *MutD* were implemented using the same wiki layout. Although, this is a valid alternative to design the website, in both cases the contents provided are limited. Notably, both websites are single page, which contains the title of the tool and a description paragraph.

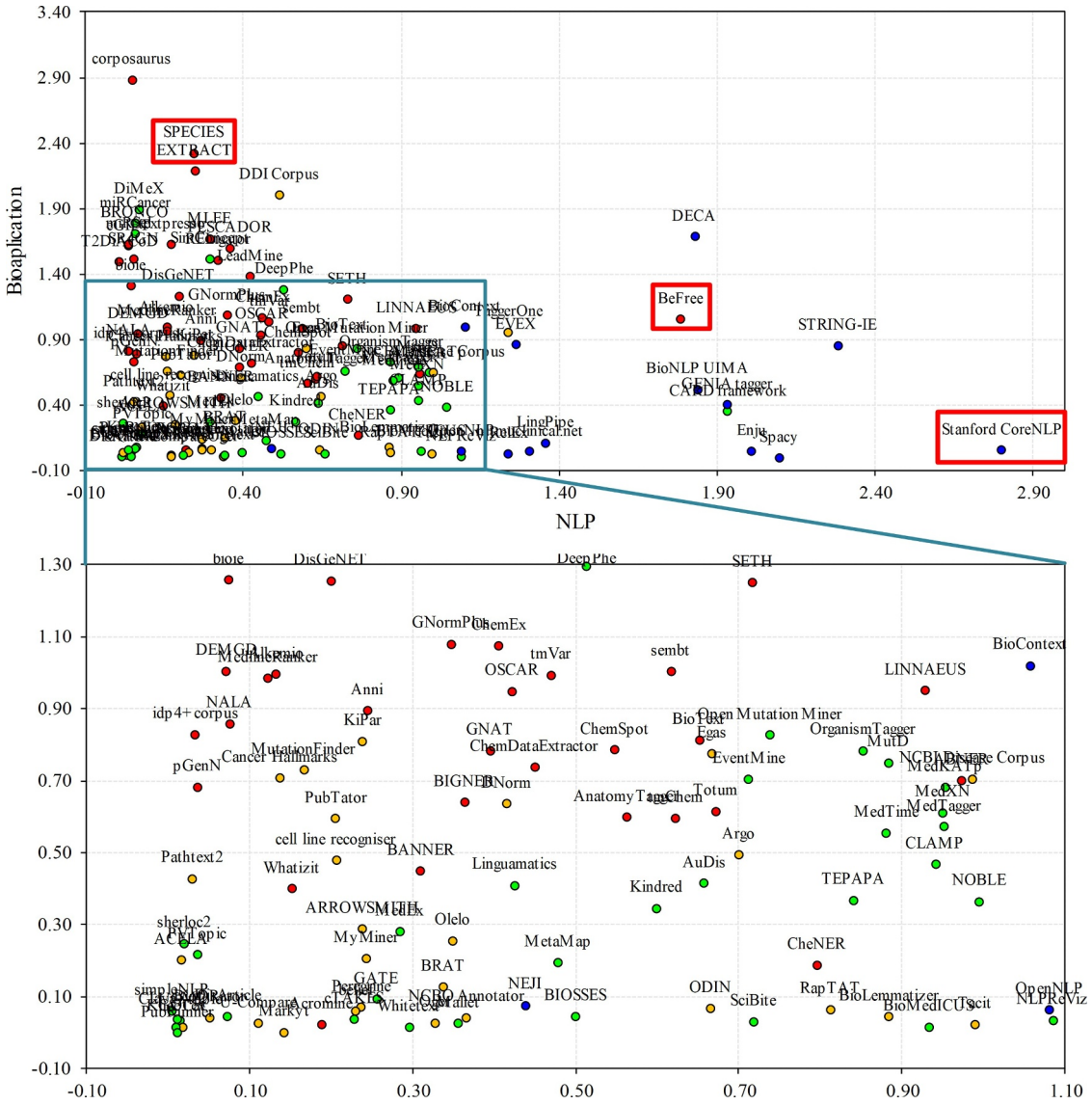

Fig. 11. Distribution of BioTM tools according to the use of terminology related to NLP (*X* axis) and Biomedical applications (*Y* axis). Colour denotes cluster assignment. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

In the case of the *MPTM* tool, it was observed the tool name and its description are presented using an image without meta-information, which hinders readability by automatic processing algorithms.

6. Conclusions

Nowadays, it is common practice to disseminate software development via websites, either websites specialised in the software itself or the websites of the originating project. This is true for both academia and industry, especially for those projects advocating the production of open-source and free software.

The main contribution of the proposed methodology is to enable the comprehensive evaluation of online contents and the alignment of these contents with supporting literature. It can be applied to any software domain, and takes advantage of existing knowledge resources and public access to literature. Its main goal is to assess how software relate to each other, considering the concepts highlighted in online descriptions. It is thus possible to identify software with similar features or goals as well as tasks that lack sufficient development. Also, this methodology helps uncover differences between online and literature contents, which are relevant to understand and enhance software search indexing and recommendation.

In the future, this methodology will be extended to gain insights into the natural evolving of website contents and literature. This knowledge is of interest to fine tune the update of website contents to keep attracting potential users and maintain existing

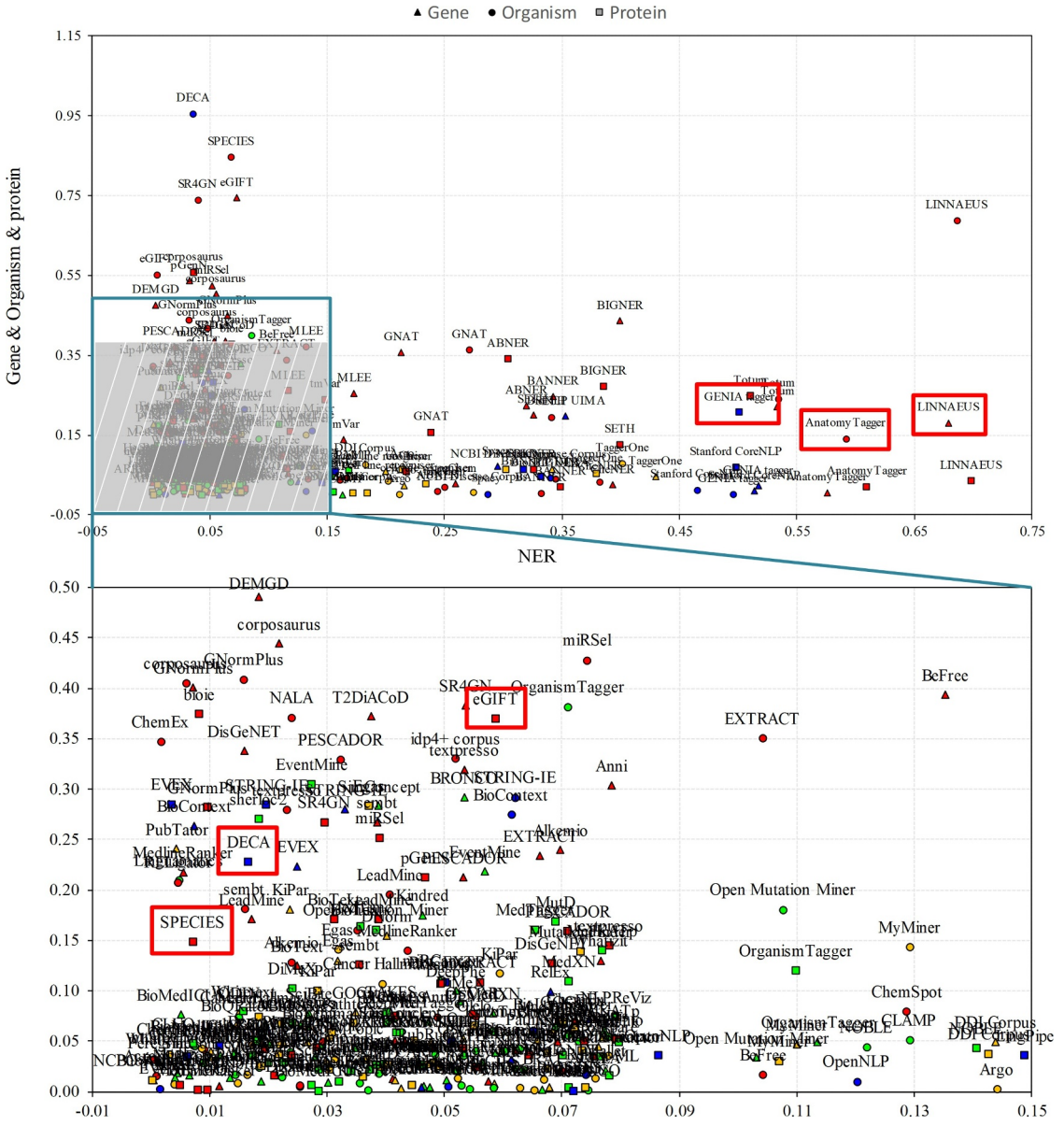

Fig. 12. Distribution of BioTM tools in terms of mentions to the concept 'NER' (*X* axis) and the biomedical entities 'Gene', 'Organism' and 'Protein' (*Y* axis). Colour denotes cluster assignment. The shaded area highlights points with poor incidence of these concepts. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 1
BioTM tools with the highest and lowest online visibility scores.

Tool	***Sim*** $(\vec{s}, \vec{a})$	***Tech*** $(\vec{s})$	***Dom*** $(\vec{s})$	Online visibility
OrganismTagger	0.697	0.793	0.835	0.697
DDI Corpus	0.942	0.534	0.844	0.696
ChemSpot	0.699	0.729	0.873	0.690
MutD	0	0.284	0	0.085
MPTM	0	0.283	0	0.084
MedXN	0	0.250	0	0.075

software engagement. It is also important to perceive new software trends and discriminate between those motivated by new application requirements and those emerging from technical novelty. Finally, efforts will be invested in linking software with applications, in particular the availability of supporting resources or software extensions, which is desirable but often missing in software websites.

Acknowledgements

SING group thanks CITI (*Centro de Investigación, Transferencia e Innovación*) from University of Vigo for hosting its IT infrastructure.

This work was partially supported by the Portuguese Foundation for Science and Technology (FCT) under the scope of the strategic funding of UID/BIO/04469/2013 unit and COMPETE 2020 (POCI-01-0145-FEDER-006684). The authors also acknowledge the Ph.D. grants of Martín Pérez-Pérez and Gael Pérez-Rodríguez, funded by the Xunta de Galicia.

Supplementary materials

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.ipm.2018.11.011.

References

Adamo, F., Attivissimo, F., Di Nisio, A., & Spadavecchia, M. (2015). An automatic document processing system for medical data extraction. *Measurement, 61*, 88–99. https://doi.org/10.1016/J.MEASUREMENT.2014.10.032.

Amer, E. (2015). Enhancing efficiency of web search engines through ontology learning from unstructured information sources. *2015 IEEE international conference on information reuse and integration* (pp. 542–549). . IEEE https://doi.org/10.1109/IRI.2015.87.

Bernstein, J. (2018). *Apache lucene core*.

Biomedical Linked Annotation Hackathon 3. (2017). (2017). Retrieved January 23, 2018, from http://blah3.linkedannotation.org/.

Biomedical Text Mining Software Tools and Databases. (2018). Retrieved February 6, 2018, from https://omictools.com/text-mining-category.

Chien, Y.-C., Liu, M.-C., Wu, T.-T., Lai, C.-H., & Huang, Y.-M. (2016). Enriching search queries to construct comprehensive concept maps for online inquiries: a case study of a food web. *Journal of Internet Technology, 17*(1), 19–27. Retrieved from http://jit.ndhu.edu.tw/ojs/index.php/jit/article/view/1220.

Cohen, K. B., Demner-Fushman, D., Ananiadou, S., & Tsujii, J.-I. (2017). Biomedical natural language processing in 2017: The view from computational linguistics.

Entrez Utilities Web services. (2010). Retrieved December 3, 2018, from https://www.ncbi.nlm.nih.gov/books/NBK25501/.

Fernández-Reyes, F. C., Hermosillo-Valadez, J., & Montes-y-Gómez, M. (2018). A Prospect-Guided global query expansion strategy using word embeddings. *Information Processing & Management, 54*(1), 1–13. https://doi.org/10.1016/J.IPM.2017.09.001.

Fleuren, W. W. M., & Alkema, W. (2015). Application of text mining in the biomedical domain. *Methods, 74*, 97–106. https://doi.org/10.1016/j.ymeth.2015.01.015.

Fluck, J., & Hofmann-Apitius, M. (2014). Text mining for systems biology. *Drug Discovery Today, 19*(2), 140–144. https://doi.org/10.1016/j.drudis.2013.09.012.

Gopalakrishnan, T., Sengottuvelan, P., Bharathi, A., & Lokeshkumar, R. (2018). An approach to webpage prediction method using variable order Markov model in recommendation systems. *Journal of Internet Technology, 19*(2), 415–424. Retrieved from http://jit.ndhu.edu.tw/ojs/index.php/jit/article/view/1661.

Harrow, I., Filsell, W., Woollard, P., Dix, I., Braxenthaler, M., Gedye, R., et al. (2013). Towards virtual knowledge broker services for semantic integration of life science literature and data sources. *Drug Discovery Today, 18*(9–10), 428–434. https://doi.org/10.1016/J.DRUDIS.2012.11.012.

Hedley, J. (2017). jsoup: Java HTML Parser. Retrieved January 1, 2018, from https://jsoup.org/.

(Computer scientist)Hofmann, M., & Klinkenberg, R. (2013). *RapidMiner : data mining use cases and business analytics applications.* Chapman and Hall/CRC.

Islamaj Dogan, R., Kim, S., Chatr-Aryamontri, A., Chang, C. S., Oughtred, R., Rust, J., et al. (2017). The BioC-BioGRID corpus: Full text articles annotated for curation of protein-protein and genetic interactions. *Database : The Journal of Biological Databases and Curation, 2017*. baw147 https://doi.org/10.1093/database/baw147.

Jiang, Y., Bai, W., Zhang, X., & Hu, J. (2017). Wikipedia-based information content and semantic similarity computation. *Information Processing & Management, 53*(1), 248–265. https://doi.org/10.1016/J.IPM.2016.09.001.

Kassing, S., Oosterman, J., Bozzon, A., & Houben, G.-J. (2015). Locating domain-specific contents and experts on social bookmarking communities. *Proceedings of the 30th annual ACM symposium on applied computing - SAC '15* (pp. 747–752). New York, USA: ACM Press. https://doi.org/10.1145/2695664.2695777.

Kaushik, S., Baloni, P., & Midha, C. K. (2018). Text mining resources for bioinformatics. *Reference Module in Life Sciences*. https://doi.org/10.1016/B978-0-12-809633-8.20499-8.

Krallinger, M. (2006). BioCreative - Bio-NLP tools. Retrieved February 26, 2018, from http://biocreative.sourceforge.net/bionlp_tools_links.html.

Krallinger, M., Leitner, F., Rabal, O., Vazquez, M., Oyarzabal, J., & Valencia, A. (2015a). CHEMDNER: The drugs and chemical names extraction challenge. *Journal of Cheminformatics, 7*(suppl 1)https://doi.org/10.1186/1758-2946-7-S1-S1.

Krallinger, M., Rabal, O., Leitner, F., Vazquez, M., Salgado, D., Lu, Z., et al. (2015b). The CHEMDNER corpus of chemicals and drugs and its annotation principles. *Journal of Cheminformatics, 7*(Suppl 1)https://doi.org/10.1186/1758-2946-7-S1-S2.

Krallinger, M., & Alfonso, V. (2017). BioCreative V.5 challenge evaluation workshop. *Fundación CNIO Carlos III*.

Lamurias, A., & Couto, F. M. (2018). Text mining for bioinformatics using biomedical literature. *Reference Module in Life Sciences*. https://doi.org/10.1016/B978-0-12-809633-8.20409-3.

McEntyre, J., & Lipman, D. (2001). PubMed: Bridging the information gap. *CMAJ : Canadian Medical Association Journal, 164*(9), 1317–1319.

Ms, S., Kumar, P., & Mukesh, R. (2017). Automatic extraction of domain specific hidden data for efficient response by search engine. *International Journal of Research and Engineering, 4*(4), 130–132. Retrieved from http://digital.ijre.org/index.php/int_j_res_eng/article/view/275.

Pant, G., & Pant, S. (2018). Visibility of corporate websites: The role of information prosociality. *Decision Support Systems, 106*, 119–129. https://doi.org/10.1016/J.DSS.2017.12.006.

Pérez-Pérez, M. (2017). BeCalm - NER resources. Retrieved January 1, 2018, from http://www.becalm.eu/NerResources.

Ramos, J (2003). Using TF-IDF to Determine Word Relevance in Document Queries. Retrieved from https://pdfs.semanticscholar.org/b3bf/6373ff41a115197cb5b30e57830c16130c2c.pdf.

Salton, G., & Buckley, C. (1988). Term-weighting approaches in automatic text retrieval. *Information Processing & Management, 24*(5), 513–523. https://doi.org/10.1016/0306-4573(88)90021-0.

Shannon, P., Markiel, A., Ozier, O., Baliga, N. S., Wang, J. T., Ramage, D., et al. (2003). Cytoscape: A software environment for integrated models of biomolecular interaction networks. *Genome Research, 13*(11), 2498–2504. https://doi.org/10.1101/gr.1239303.

Singhal, A., Leaman, R., Catlett, N., Lemberger, T., McEntyre, J., Polson, S., et al. (2016). Pressing needs of biomedical text mining in biocuration and beyond: Opportunities and challenges. *Database : The Journal of Biological Databases and Curation, 2016*https://doi.org/10.1093/database/baw161.

Wang, Q., S Abdul, S., Almeida, L., Ananiadou, S., Balderas-Martínez, Y. I., Batista-Navarro, R., et al. (2016). Overview of the interactive task in BioCreative V. *Database : The Journal of Biological Databases and Curation, 2016*https://doi.org/10.1093/database/baw119.

Xuan, J., Luo, X., Zhang, G., Lu, J., & Xu, Z. (2016). Uncertainty analysis for the keyword system of web events. *IEEE Transactions on Systems, Man, and Cybernetics: Systems, 46*(6), 829–842. https://doi.org/10.1109/TSMC.2015.2470645.

Yan, Y., Liu, G., Wang, S., Zhang, J., & Zheng, K. (2017). Graph-based clustering and ranking for diversified image search. *Multimedia Systems, 23*(1), 41–52. https://doi.org/10.1007/s00530-014-0419-4.

Yang, H.-C., Hsiao, H.-W., & Lee, C.-H. (2011). Multilingual document mining and navigation using self-organizing maps. *Information Processing & Management, 47*(5), 647–666. https://doi.org/10.1016/J.IPM.2009.12.003.

Yang, P., Gao, W., Tan, Q., & Wong, K.-F. (2013). A link-bridged topic model for cross-domain document classification. *Information Processing & Management, 49*(6), 1181–1193. https://doi.org/10.1016/J.IPM.2013.05.002.

Yunzhi, C., Huijuan, L., Shapiro, L., Travillian, R. S., & Lanjuan, L. (2016). An approach to semantic query expansion system based on Hepatitis ontology. *Journal of Biological Research (Thessalonike, Greece), 23*(suppl 1), 11. https://doi.org/10.1186/s40709-016-0044-9.

Zeng, Z., Shi, H., Wu, Y., & Hong, Z. (2015). Survey of natural language processing techniques in bioinformatics. *Computational and Mathematical Methods in Medicine, 2015*, 1–10. https://doi.org/10.1155/2015/674296.

Zhang, H., Wang, D., Wang, L., Bi, Z., & Chen, Y. (2014). A semantics-based method for clustering of Chinese web search results. *Enterprise Information Systems, 8*(1), 147–165. https://doi.org/10.1080/17517575.2013.857793.